Daffodil International University

Department of Software Engineering, FSIT

SWE-431 Project / Thesis

Project Documentation

Utility management system

Supervised by:

Ms. Syeda Sumbul Hossain

Senior Lecturer

Department of Software Engineering

Daffodil International University

Submitted by:

Mobashira Farjana

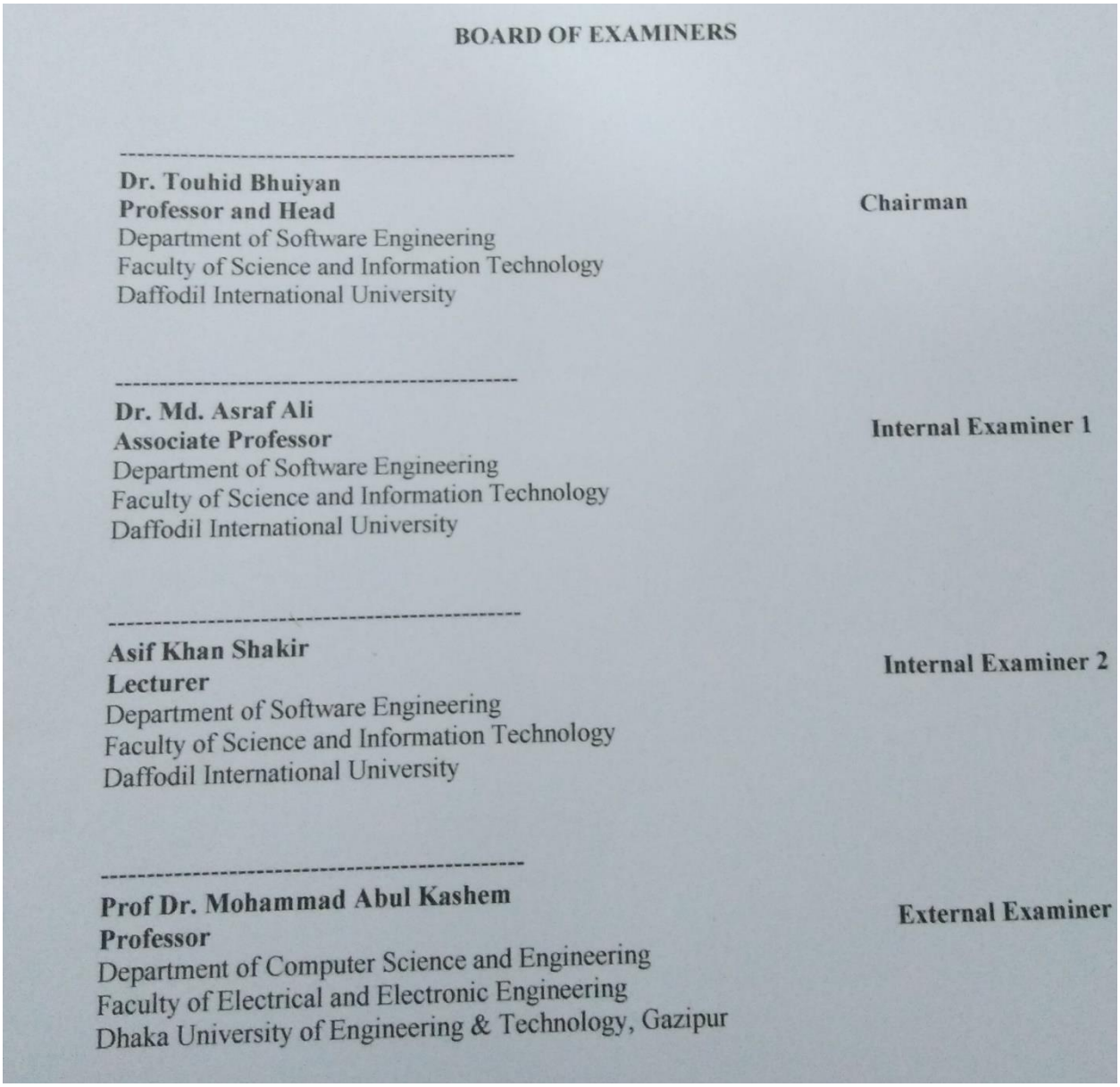
Id: 152-35-1175

Department of Software Engineering

Daffodil International University

Approval

This project titled "**Utility Management System**", submitted by Mobashira Farjana(152-35-1175) to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc in Software Engineering and approved as to its style and contents.



BOARD OF EXAMINERS

Dr. Touhid Bhuiyan
Professor and Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman

Dr. Md. Asraf Ali
Associate Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Asif Khan Shakir
Lecturer
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

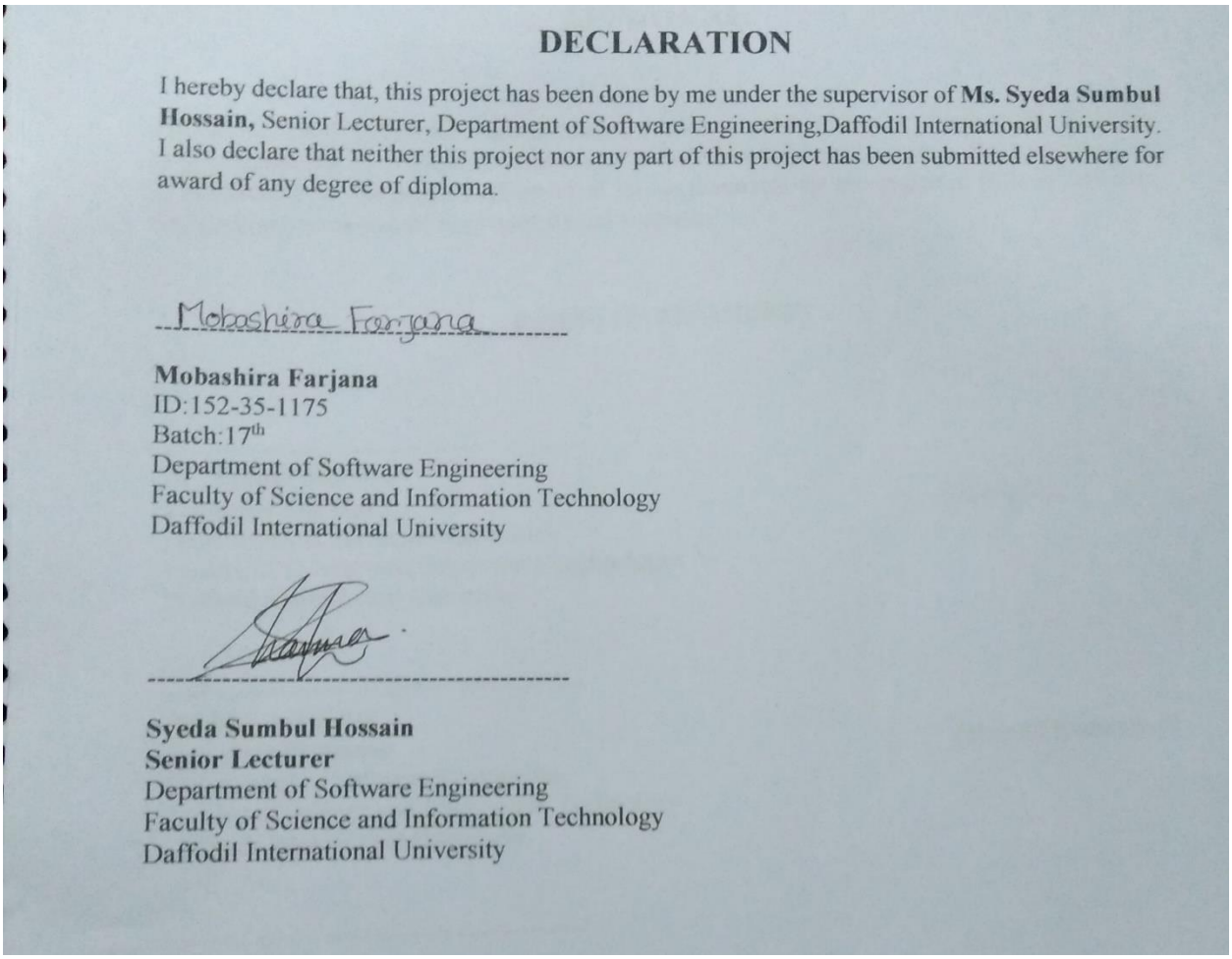
Internal Examiner 2

Prof Dr. Mohammad Abul Kashem
Professor
Department of Computer Science and Engineering
Faculty of Electrical and Electronic Engineering
Dhaka University of Engineering & Technology, Gazipur

External Examiner

DECLARATION

I hereby declare that, this project has been done by me under the supervisor of **Ms. Syeda Sumbul Hossain,** Senior Lecturer, Department of Software Engineering,Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree of diploma.



DECLARATION

I hereby declare that, this project has been done by me under the supervisor of **Ms. Syeda Sumbul Hossain,** Senior Lecturer, Department of Software Engineering,Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree of diploma.

Mobashira Farjana

Mobashira Farjana
ID:152-35-1175
Batch:17th
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Syeda Sumbul Hossain
Senior Lecturer
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Acknowledgement

At first, I would like to thank almighty Allah for giving me the proper time to successfully complete the project.

I am very grateful to my parents they are also an important inspiration for me. So that with due respect, I express my gratitude to them.

I can't express enough thanks my honorable supervisor "Ms. Syeda Sumbul Hossain" Faculty of software Engineering, Daffodil International University for her care, support and also excellent guidance having trust in my capability to complete this project.

I want to express our heartiest gratitude to Head, Department of swe, for encouraging me and providing me with such an opportunity.

.

Table of Contents

List of Figures

.

Chapter-01 Introduction

1.1 Introduction:

Through the system, user (householder) can easily view bills, bill type, daily cost and also view total cost. This section gives an overview of everything included in this project and discusses some of the benefits.

1.2 Purpose:

Through this document, the project "Utility Management system" will get a good idea. This document is main purpose is given a good idea for user and their work. This document will highlight the benefits of the project. This document will discuss the requirements, design, implementation and testing.

1.3 Project overview:

Today's technology is getting update that has made our life easier and quicker. Some people store their utility bills manually that's why one or more bill missed to pay and do not pay bills before dateline. To keep those information need to have an application. In here user can added bills edit bills information as they require.

1.4 Problem Statement:

Sometimes we forget to pay all utility bills in time. For this reason often dateline is over. To avoid this problems, do help my project.

1.5 Goals:

The main goals of this project are to the web based application. Most people are now depend on online. Some people search that type application makes easy to maintain their utility system.
So I think this project (utility management system) will be useful for many people.

1.6 Scope:

Through this system you can easily add your utility bills, update bills, check bills and also view your daily cost and total cost.

1.7 Stakeholder:

There is one type stakeholder such as:

- Housekeeper.

Brief descriptions about stakeholder are given below.

Housekeeper:

Can view utility bill type, monthly cost, total cost. update bills, check bills and also added daily cost.

1.8 Project Schedule:

Table 1.8: Project schedule

Activities	**Duration (in week)**	**Total week**
Brainstorming	Week-1, Week-2	2
Problem identification	Week-2, Week-3	2
Requirement analysis	Week-4	1
Sketching	Week-5	1
Design specification	Week-6, Week-7	2
Database design	Week-8	1
Implementation	Week-9, Week-10 Week-11, Week-12	4
Testing	Week-13, Week-14	3
Delivery	Week-15, Week-16	1

1.9 Release Plan

The release plan is given below:

Table 1.9: Release plan

Version	**Feature**	Date
V0.1	System design, login	05-08-2019
V0.2	View bills, bill type, add bill, daily cost, total cost.	30-11-2019

Chapter 2: System Analysis

2.1 Requirement Analysis

Requirement analysis is a process of help user needs for application. Requirement analysis involves the task that decides the needs of the software, which mainly incorporates objections and requirements of different customer. Requirement analysis needs to the good quality of the application.

2.2 Requirement Gathering

There are many strategies for collecting requirement gathering, such as brainstorming, questionnaires, interview and customer observation.

I first meet the supervisor and made a list of the requirement. I've talked about my project with different users and I've got work from them.

2.3 Feature

- Password protected login system.
- Householder view bills item
- Householder add bills item.
- Householder edit bills item.
- Householder check bill item.
- Householder added daily cost.
- Householder view total cost.

2.4 Requirement Specification

2.4.1 Functional Requirement

Functional requirements are mandatory for the any system. For this system functional requirements are mentioned below.

Table 2.5.1.1: Login User

ID: FR-01
Requirement Name: Login User
Description: User can input valid email and password then user successfully login.
Stockholders: Housekeeper

Table 2.5.1.2: Logout User

ID: FRQ-02
Requirement Name: Logout User
Description: If the user is logged out then user session detail clear.
Stockholders: Housekeeper

Table 2.5.1.3: User registration

ID: FRQ-03
Requirement Name: User registration.
Description: User input valid information then user successfully registered
Stockholders: Housekeeper

Table 2.5.1.4: View bills item

ID: FRQ-04
Requirement Name: view bills item
Description: if householder want to view item they can see.
Stockholders: Housekeeper

Table 2.5.1.5 Add bills item

ID: FRQ-05
Requirement Name: add bills item
Description: householder can added bills item.
Stockholders: Housekeeper.

Table 2.5.1.6 check bills item

ID: FRQ-06
Requirement Name: check bills item
Description: housekeeper can check bills item.
Stockholders: Housekeeper.

Table 2.5.1.7 add daily cost

ID: FRQ-07
Requirement Name: check bills item.
Description: householder can add daily cost.
Stockholders: Housekeeper.

Table 2.5.1.8 view total cost

ID: FRQ-08
Requirement Name: view total cost
Description: householder can view total cost from this system
Stockholders: Housekeeper.

2.5.2 Non-Functional Requirement

For this system non-functional requirements are mentioned below.

Table 2.5.2.1: Privacy

ID: NFRQ-1
Requirement Name: Privacy.
Description: System has to protect user data and confidential information.
Stockholders: Housekeeper.

Table 2.5.2.2: Data Capacity

ID: NFRQ-2
Requirement Name: Data Capacity
Description: This system need to handle thousands of data.
Stockholders: Housekeeper.

Table 2.5.2.3: Interface Requirement

ID: NFRQ-3
Requirement Name: Interface Requirement.
Description: System should be user friendly for all users.
Stockholders: Housekeeper.

2.6 Use case of proposed system

2.6.1 Use-case Diagram

A use case diagram is a dynamic or behavior diagram in Unified Modeling Language. Use case diagrams model the functionality of a system using actors and use cases. Use cases are a set of actions, services, and functions that the system needs to perform. In this context, a "system" is something being developed or operated, such as a web site. The "actors" are people or entities operating under defined roles within the system.

In that project use case diagram is valuable for visualizing the functional requirements of a system that will translate into design choices and development priorities.

They also help identify any internal or external factors that may influence the system and should be taken into consideration.

Use case diagram provides a good analysis from outside the system. It is also represent how that system interacts with actors without worrying about the details of how that functionality is implemented.

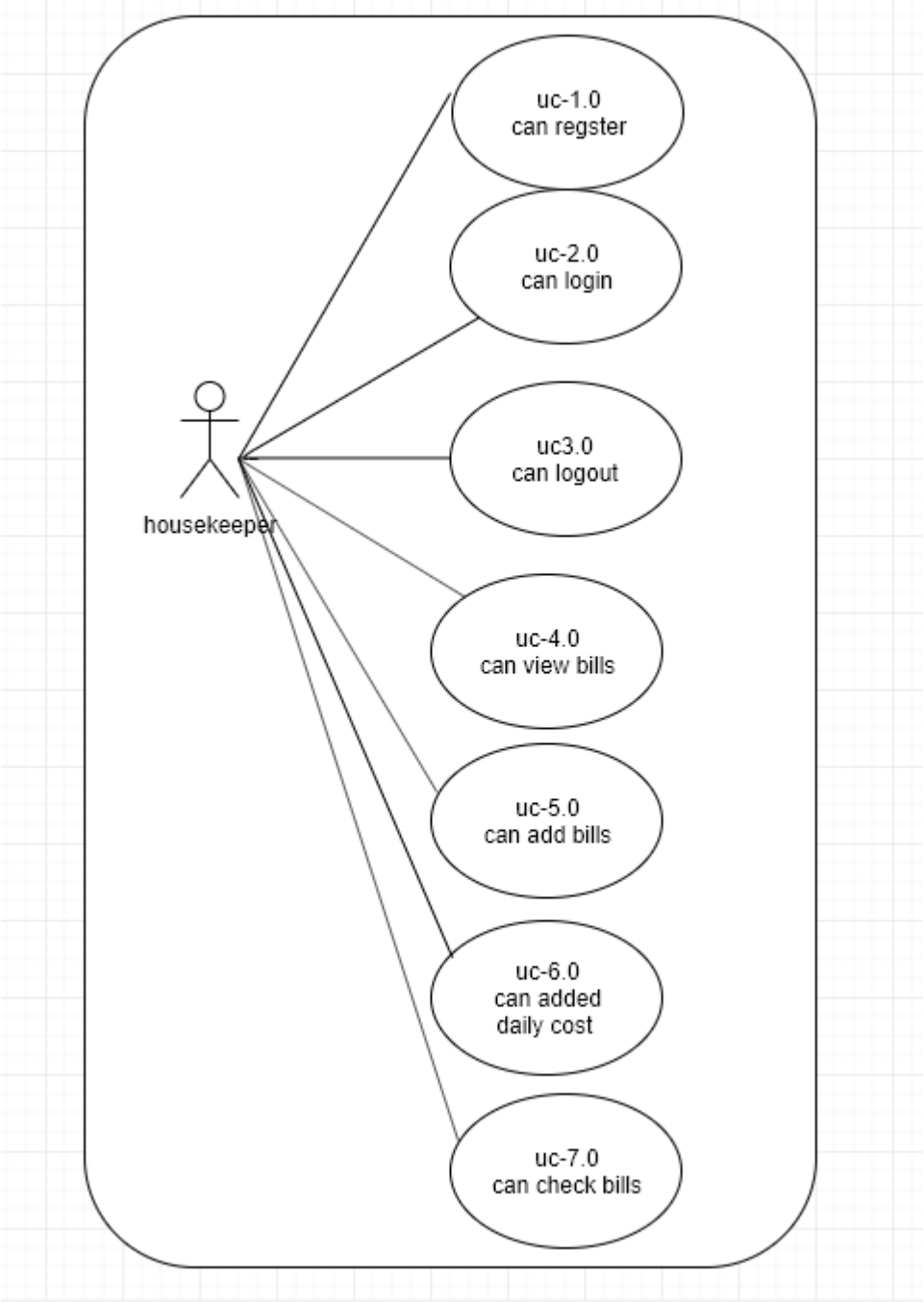


Figure 2.6.1: Use-case diagram

2.6.2 Use-Case Table

Table 2.6.2.1: Can Register

Use Case Title	Can register
Use Case ID	Uc-1.0
Preconditions	User select the register form and fill up it.
Actors	Housekeeper
Success End Condition	User successfully register
Failure End Condition	Display register error message “ ”
Description	After all requirement field fill up user can register for the system

Table 2.6.2.2: Can login

Use Case Title	Can login
Use Case ID	Uc-2.0
Preconditions	1. User must be registered. 2. User select the login form and fill up it.
Actors	Housekeeper
Success End Condition	User successfully Login.
Failure End Condition	Display login error message “ ”
Description	After the user login then user can see the homepage and many option

Table 2.6.2.3: Can logout

Use Case Title	Can logout
Use Case ID	Uc-3.0
Preconditions	User must be login.
Actors	housekeeper
Success End Condition	User successfully Logout.
Failure End Condition	User not successfully Logout.
Description	If the user is logged out then users will need to log in again to go to the user page.

Table 2.6.2.4: Can view bills

Use Case Title	Can view bills
Use Case ID	Uc-4.0
Preconditions	User should be login.
Actors	housekeeper
Success End Condition	User successfully view all types of bills
Failure End Condition	User can't successfully view all types of bills
Description	After login user view all bills information.

Table 2.6.2.5: Can add bills

Use Case Title	Can add bills
Use Case ID	Uc-5.0
Preconditions	User should be login.
Actors	Housekeeper
Success End Condition	User successfully added bills as require
Failure End Condition	User can't successfully added bills as require
Description	After all requirement field fill up user then can added bills of the system

Table 2.6.2.6: check bill items

Use Case Title	check bills
Use Case ID	Uc-6.0
Preconditions	User should be login.
Actors	housekeeper
Success End Condition	User successfully check bills
Failure End Condition	User can't successfully check bills
Description	After all requirement field fill up user then can check bills of the system

Table 2.6.2.7: added daily cost

Use Case Title	added daily cost
Use Case ID	Uc-7.0
Preconditions	Need to login first of this system
Actors	Housekeeper
Success End Condition	User successfully added daily cost as require
Failure End Condition	User can't successfully added daily cost.
Description	After all requirement field fill up user then can added daily cost of the system

Chapter 3: System Design

3.1.1 Activity Diagram

Activity diagram represents the flow from one activity to another activity. The activity can be described as an operation of the system.

The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. it deals with all type of flow control by using different elements.

In this diagram, at first householder(user) go to homepage if he has registered user, he will login this system and after than login he will done other activity like view bill item, add bills, added daily cost, and check bill item.

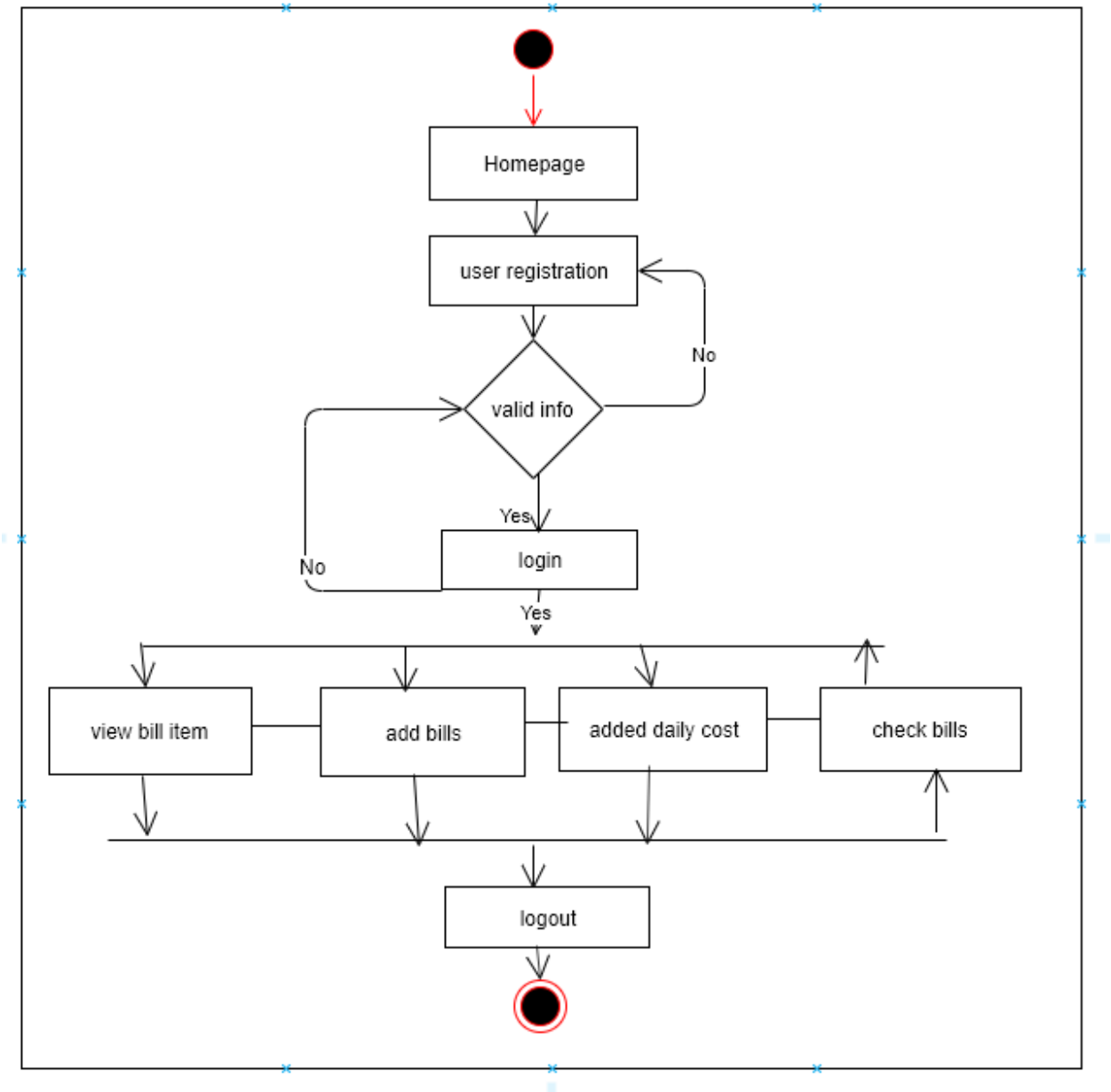


Figure 3.1.1: Activity Diagram for housekeeper

3.2.1 Dataflow Diagram (level-0)

Data flow diagrams represents the flow of data in a business information system. DFD describes the processes that are involved in a system to transfer data from the input to the file storage.

This diagram provides Logical information flow of the system, Determination of physical system construction requirements, Simplicity of notation and Establishment of manual and automated systems requirements

Following that figure we can see housekeeper input data like registration information, login, check to view item, check bill, add bill item database accept those information and shown successfully massage.

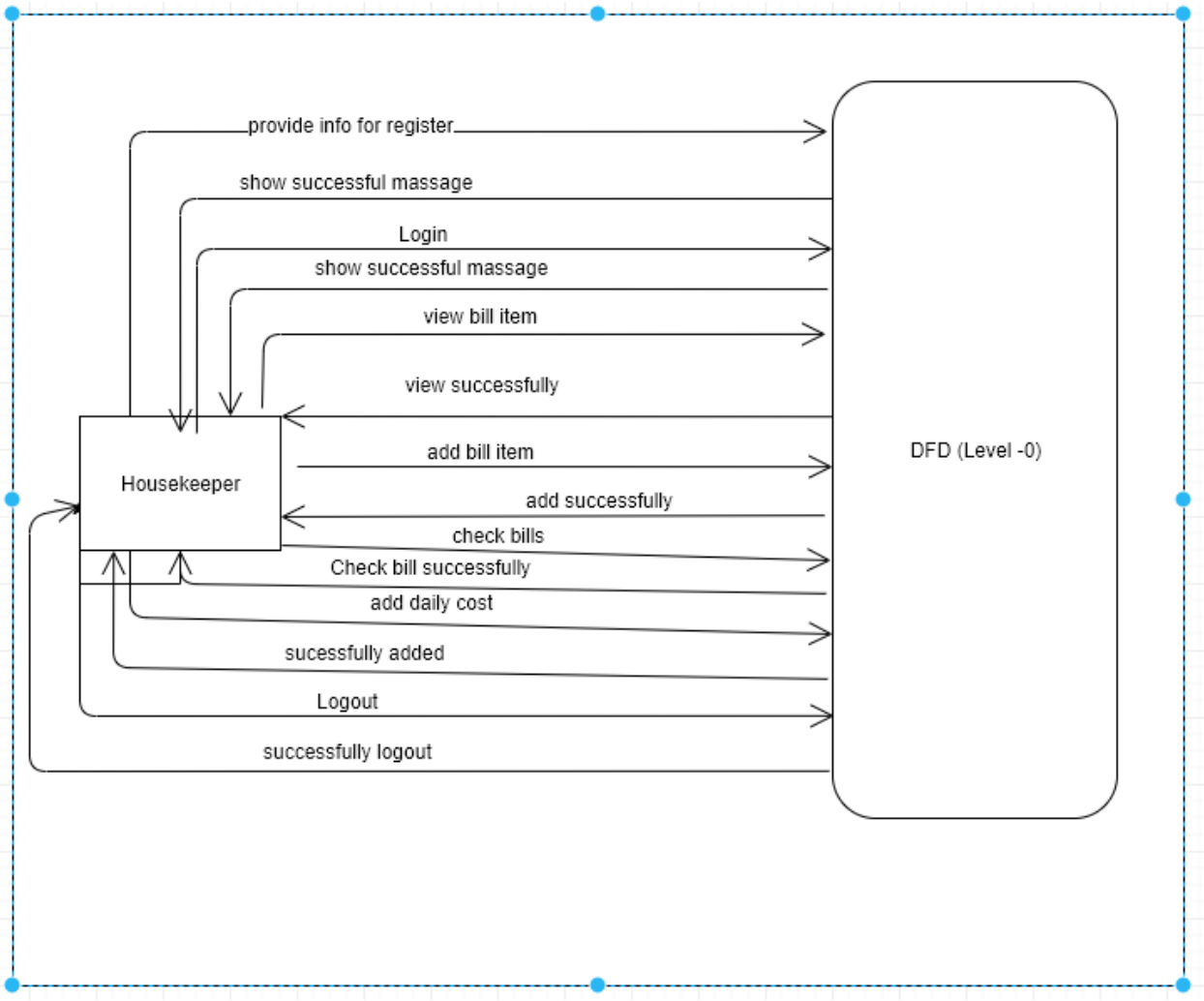


Figure 3.2.1: Dataflow diagram (Level-0)

3.3 Entity Relationship diagram

Entity relationship diagram representation of entities and their relationships to each other. Following this diagram that precisely depicting the flow of the different state of the project.

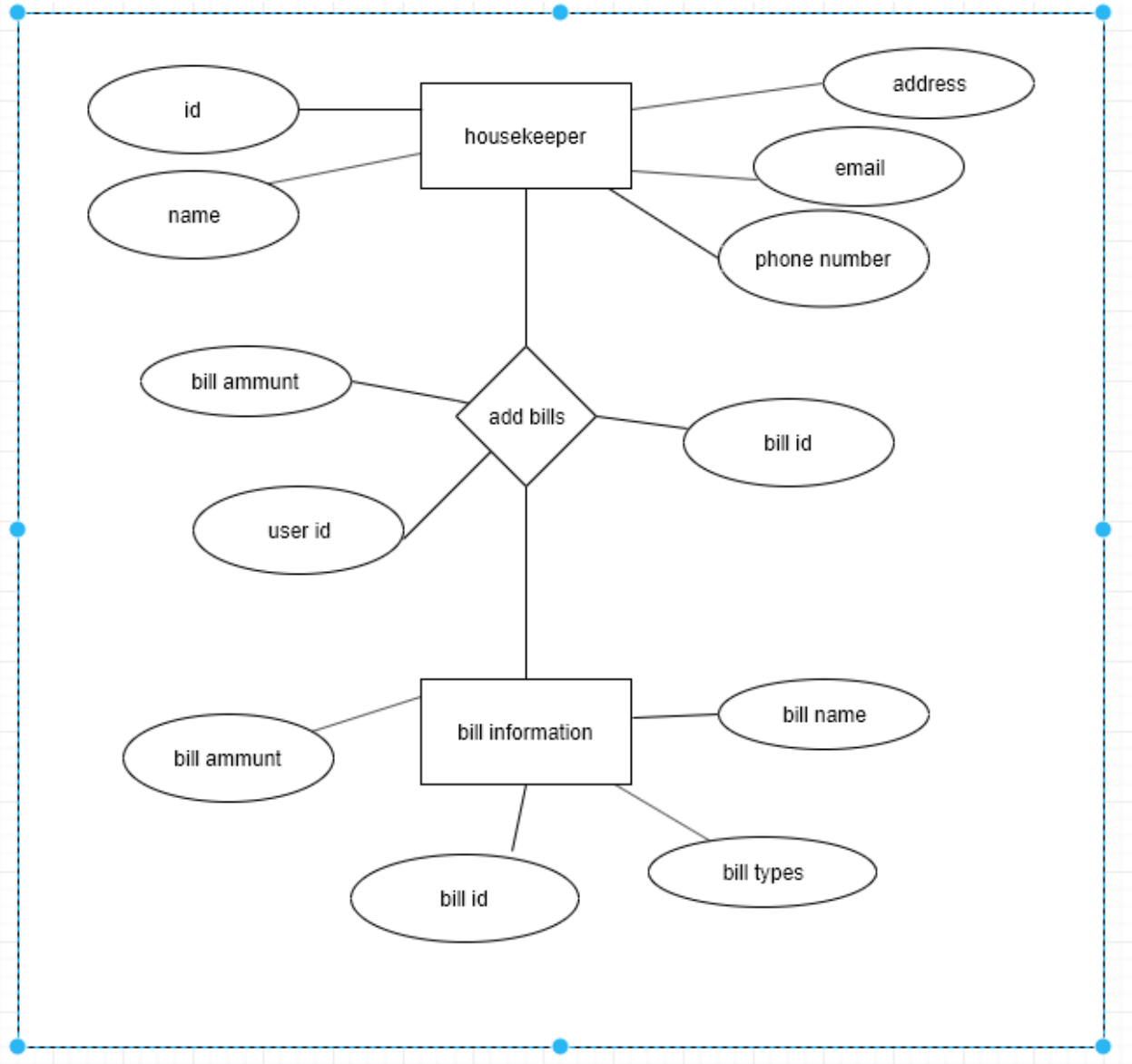


Figure 3.3: Entity Relationship diagram

Chapter 4:
System Test & Development

4.1 Introduction

This web application is a services application, this system will provide service of householder. There are only one user and some features on this system. So I think this application requires on this system. To ensure an application quality, testing need to be done. The user (householder) will give some data so be careful about the data. I've worked with that test they are given below

Functional Testing

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing

4.2 Features to be tested:

Table- 4.2: Features to be tested

Feature	Priority	Description
Login	3	User must be authenticated by login.
Logout	2	Session must be destroy after logout
Register	3	To become a user, need to be registered first.
View bill item	2	Householder must be login than View bill items.
Add bills	2	Householder should be login Then add their bills
Add daily cost	2	Housekeeper should be login first then add daily cost.

4.3 Testing Strategy:

Testing strategy determines the projects approach to testing. I will first check that requirement then I will verify requirement whether it work correctly. It is also the indicator of test that are to be performed on the whole software development life cycle. Keep the application started I will follow a few step of testing. First of all, I did the functional testing then unit testing then I did Integration testing then System testing and at last I did Acceptance testing.

4.4 Test case Table

Table 4.4.1: Login

Test case #1			Test case Name:		
System: utility Management system					
Designed by: Mobashira Farjana			Designed date: 06.06.2019		
Executed By: Mobashira Farjana			Executed Date: 07.06.2019		
Short description: If the user will be correct data input then user login successfully. Pre-conditions: • User must be registered. • Assume that, the email id is 'mobashira111@gmail.com' and password is " password "					
Step	Email	password	Expected result	Pass/fail	comment
1	Mobashira@gmail.com	Psss	Do not match our record.	pass	
2	mobashira111@gmail.com		Password field required.	Pass	
3		password	The email field is required.	pass	
4	mobashira@gmail	password	Invalid email	pass	
5	mobashira111@gmail	password	Successfully login to the application	pass	
Post-conditions: After the user login then user can see the homepage and many option.					

Table 4.4.2: Logout User

Test case #2		Test case Name: Logout User		
System: utility Management system				
Designed by: Mobashira Farjana		Designed date: 06.06.2019		
Executed By: Mobashira Farjana		Executed Date: 07.06.2019		
Short description: If the user want logout, then user will chose logout button and click the button				
Pre-conditions: User must be login				
Step	Action	Response	Pass/fail	Comment
1	Click logout from dropdown.	Successfully logout.	Pass	
2	After logout click back and reload.	Redirect to login page	Pass	
Post-conditions: Session will be destroy.				

Table 4.4.3: Registration User

Test case #3		Test case Name: Registration User		
System: utility Management system				
Designed by: Mobashira Farjana		Designed date: 08.06.2019		
Executed By: Mobashira Farjana		Executed Date: 09.06.2019		
Short description: After all requirement field fill up user can register for the system.				
Pre-conditions: User select the register form				
Step	Action	Response	Pass/fail	comment
1	All required fields are not filled yet.	Fields must not be empty.	Pass	
2	All input filed is filled up by the user.	Application will save those information	Pass	
Post-conditions: User information is saved to the database of the application program properly.				

Table 4.4.4: view bills item

Test case #4			Test case Name: view bills item	
System: utility Management system				
Designed by: Mobashira Farjana			Designed date: 08.06.2019	
Executed By: Mobashira Farjana			Executed Date: 09.06.2019	
Short description: if the housekeeper will see the bill information, they can able those item.				
Pre-conditions: housekeeper must be login.				
Step	Action	response	Fail/pass	comment
1	User information not saved database	Not found user	Fail	
2	User information saved database	Show the information.	pass	
Post-conditions: View information.				

Table 4.4.5: added bills item

Test case #5			Test case Name: added bills item	
System: utility Management system				
Designed by: Mobashira Farjana			Designed date: 08.06.2019	
Executed By: Mobashira Farjana			Executed Date: 09.06.2019	
Short description: housekeeper can add own information.				
Pre-conditions: householder must be authenticated by application.				
Step	Action	Response	Fail/pass	Comment
1	All required fields are not filled yet.	Fields must not be empty	Fail	
2	All input filed is filled up by the pet owner	Application will save those information	pass	
Post-conditions: View the doctor information.				

4.5 Development

4.5.1 Tools and Technology

This system is a web application. Tools and technology I used to build this software are alluded below.

- **Presentation Layer:** HTML5, CSS3, Bootstrap 4, JavaScript, JQuery.
- **Application Layer**: PHP, Laravel 5.5
- **Data Layer:** MySQL
- **Tools:** Sublime Text 3, Xampp 3.2.4, composer 1.9.1.

4.5.2 Reason behind choosing

- **HTML5:** HTML5 used to make structure of web page.
- **Bootstrap with CSS:** Bootstrape is a css framework and language CSS is a styling.
- **JavaScript with JQuery:** To make presentation layer interactive JavaScript is used and JQuery is its library. PHP: PHP is probably the most popular scripting language on the web. It is used to enhance web pages.
- **Laravel with PHP:** Laravel is a framework of PHP which uses MVC architecture and used for connecting presentation layer with data layer
- **MySQL:** MySQL is a database to manage and manipulate data to make connection with application layer.
- **Sublime Text and Xampp:** Sublime is an IDE or text editor uses for coding. And xampp is a control panel to start database server.

Chapter 5: Risk Management

5.1 Software Risk identification

Risk identification is a process that helps keep our project quality better and reduces the risk of a business environment. Determining the risk for a business organization is very important. When determining the risk, we must evaluate the risk very accurately so that there is no error. To find out the risk score have to know risk probability and risk impact.

Probability of the risk categorized

Probability of the risk		
Certain	5	The risk is almost certain to occur (greater than 80% chance)
Likely	4	he risk is more likely to occur than not (between 51% and 80% chance)
Possible	3	The risk is fairly likely to occur (between 21% and 50% chance)
Unlikely	2	The risk is unlikely but not impossible to occur (between 6% and 20% chance)
Rare	1	The risk is unlikely to occur (<5% chance)

Impact of the risk categorized:

Impact of the risk	
Catastrophic	5
Major	4
Moderate	3
Minor	2
Negligible	1

5.2 Software risk Register

Table 5.2: Software risk register

Risk id	Risk Description	impact	probability	Risk score
1	Technical Problem	2	3	6
2	System that crash	4	2	8
3	Information security	3	3	9
4	Misunderstand requirement	4	4	16
5	Decision are low quality	2	3	6
6	Decision delays impact project	3	2	6

5.2.1 Software Risk Matrix

Based on this risk register, we can draw a risk matrix where, Probability on the y-axis Impact on the x-axis Risk will be set in matrix according to their score.

Table 5.2.1: Software risk matrix

Probability/Impact	Negligible(1)	Minor(2)	Moderate(3)	Major(4)	Catastrophic(5)
Certain(5)					
Likely(4)			6	4	
Possible(3)		1,5	3		
Unlikely(2)				2	
Rare(1)					

15-25 Extreme Risk

8-12 High Risk

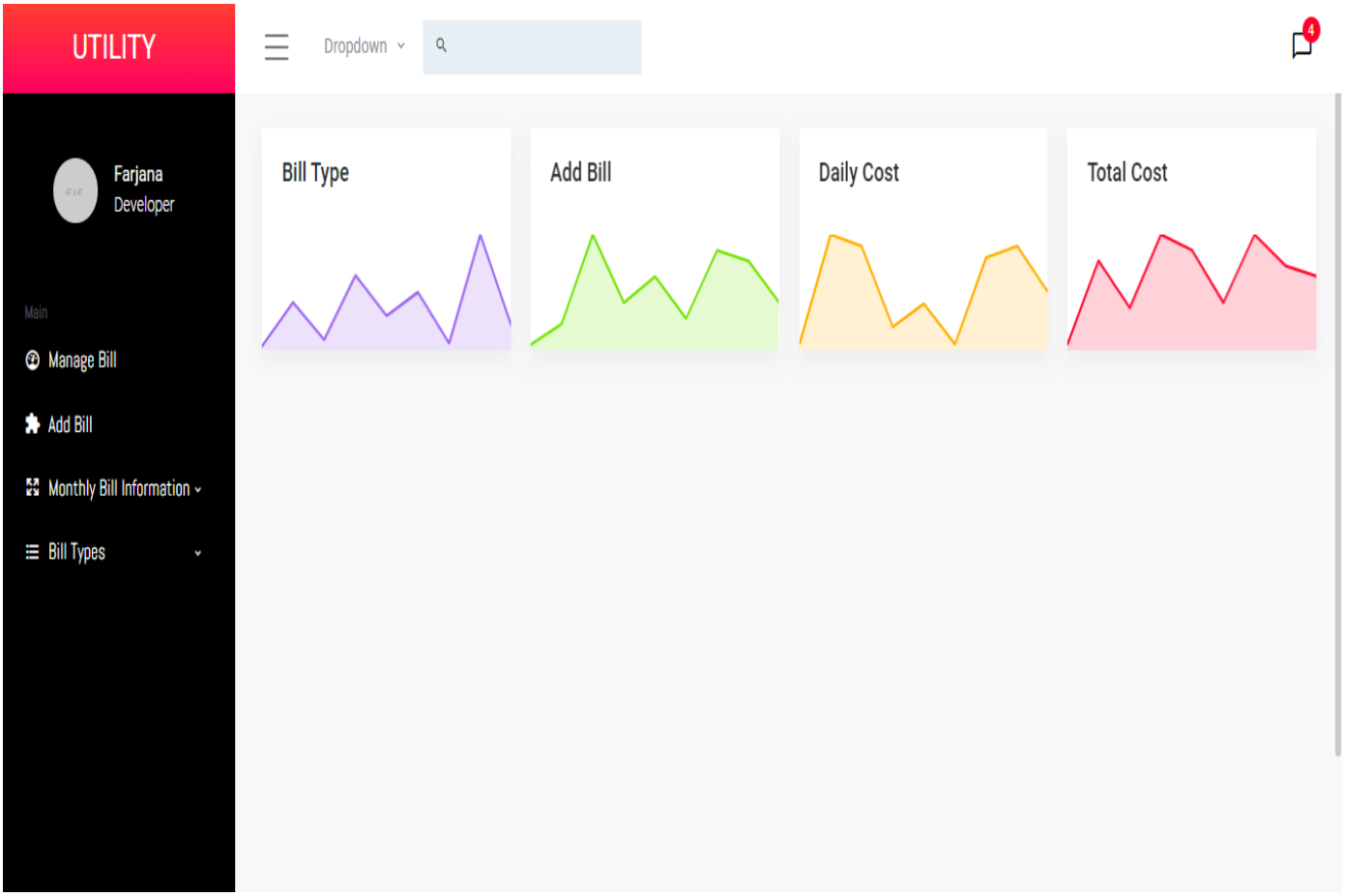
4-6 Moderate Risk

1-3 Low Risk

Chapter 6: User Manual

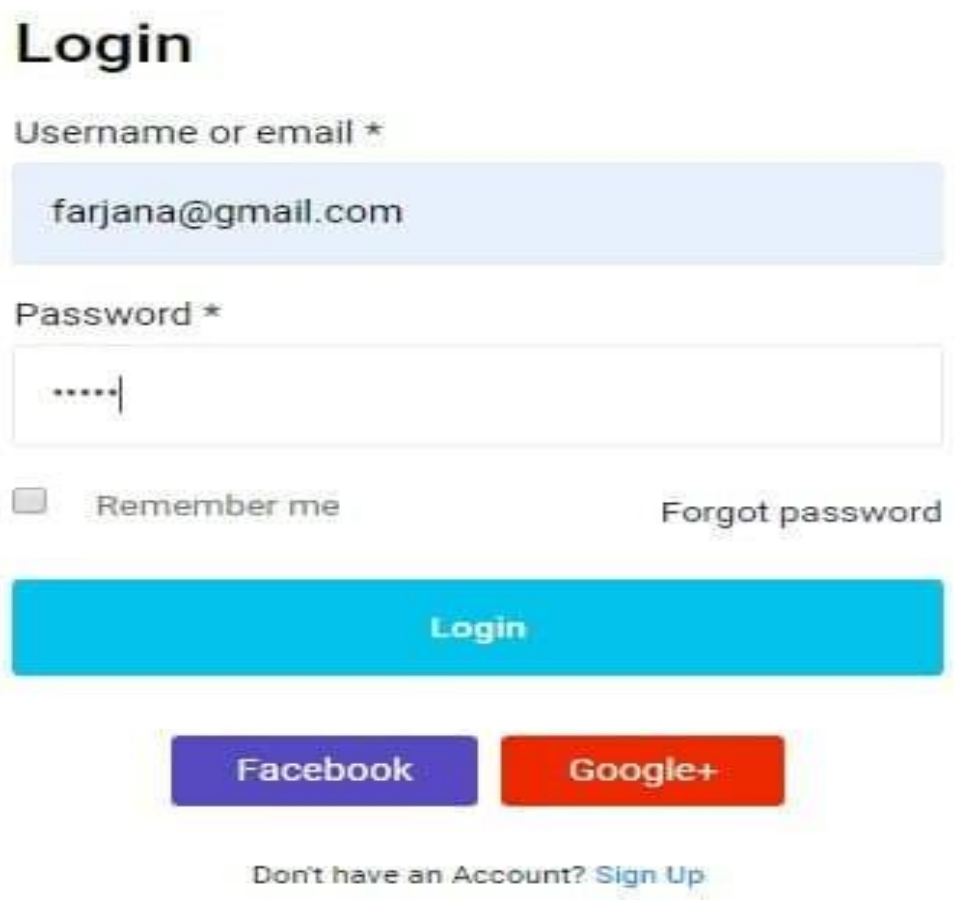
6.1 Home page

After provide correct username and password householder (user) can show this home page.



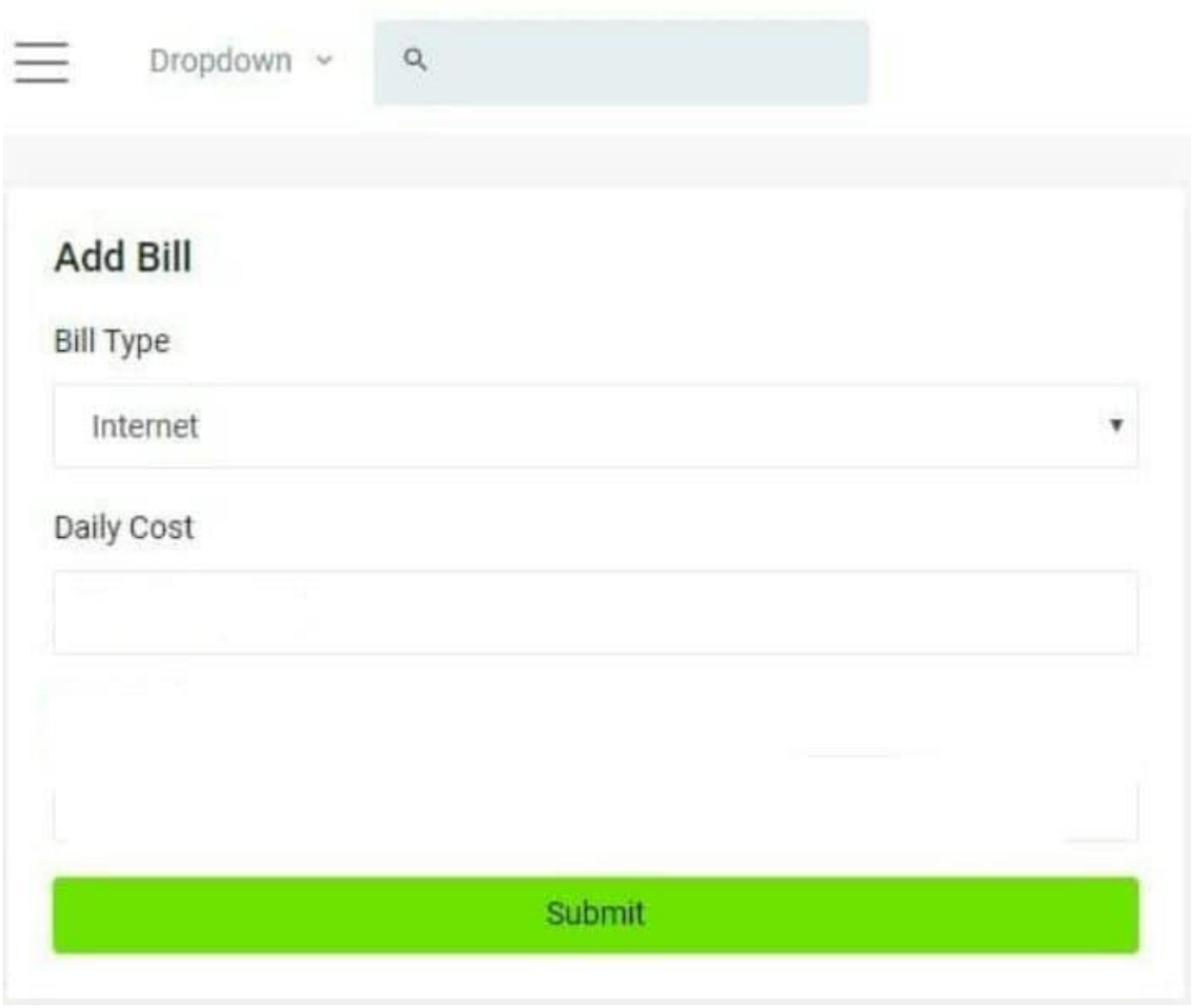
6.2 Login page

To enter utility management system user (housekeeper) need to authentic him/herself first. User(housekeeper) should provide valid username and password to use that system.



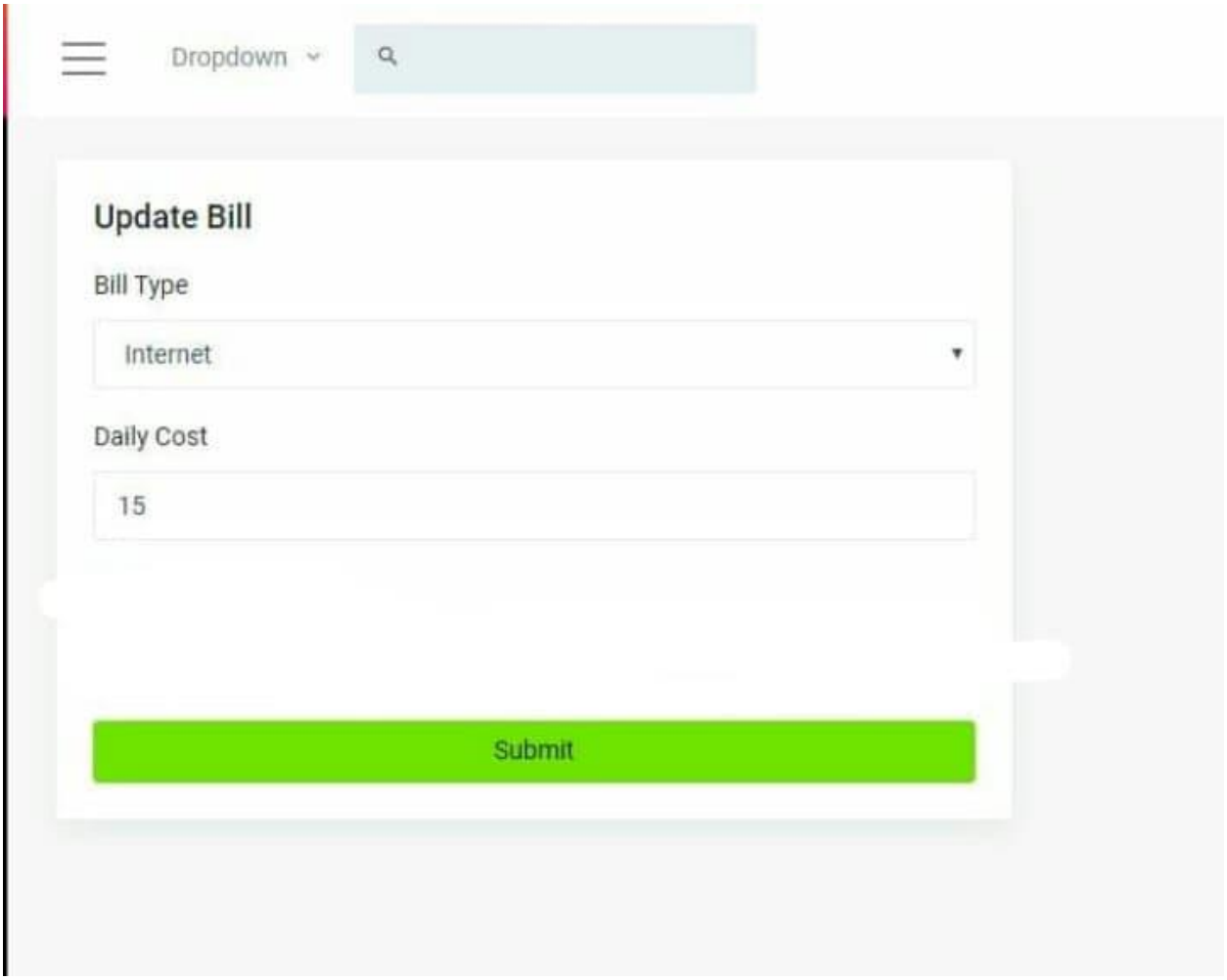
6.3 add bill

After login, user (Housekeeper) can easily access this system. View bill item and select one. Also added bill amount and submitted that record.



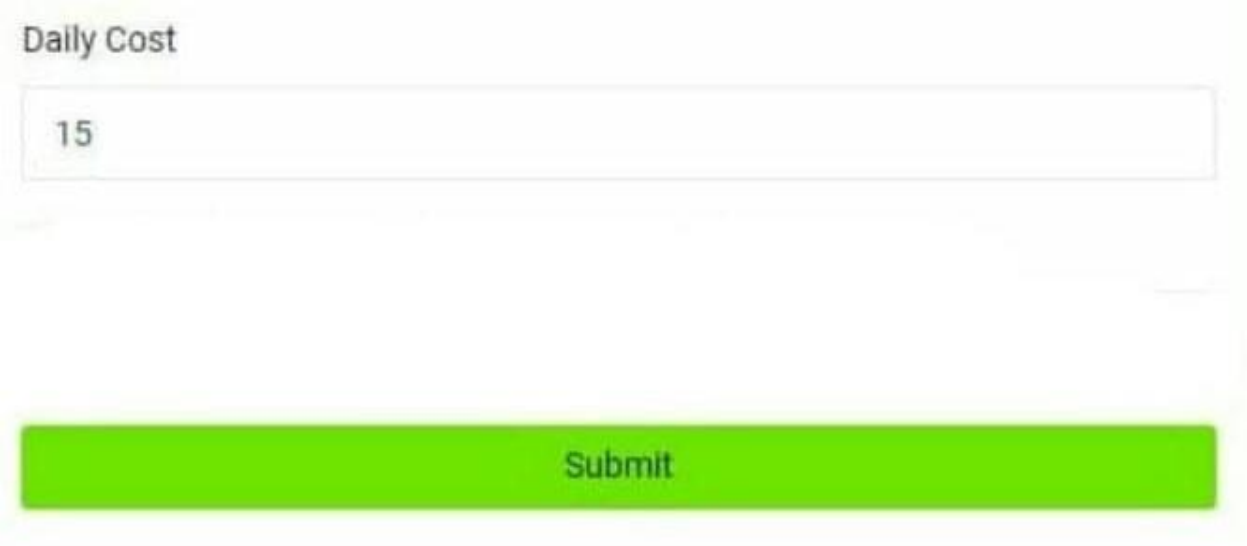
6.4 update bill

Housekeeper(user) can update bill item as he/she require.



6.5 daily cost

Housekeeper (user) added his/her daily cost as he requires



Chapter 7: Conclusion

7.1 Github Link: https://github.com/mubashira-farjana/utility

7.2 Project Summary:

I have started the project work from May. I have first discuss to my supervisor then I have done list down the features. After confirming the features then I started the system diagram and database design. After ensure the diagram the diagram and system design of my project then I started the coding part.

After complete the coding part then I am started the testing part. To started testing work I have been taken some idea of testing and that's my supervisor helped me.

7.3 Limitation

I have tried my best to make the application better and fulfill requirements but some of them were not possible make for time shortage.

It is difficult to develop something without any limitation. This project has some limitations. Follow as -

- Not fully responsive
- Not highly secure
- User's password are not encrypted

7.4 Future Scope

I think I learned a lot through this project. I would thank them that all the people give me such a nice opportunity and helped me. I think this experience will help me a lot in the future and will support for creating new features.